Evaluation of Multilingual Parts of Speech Tagging

Shubham Milind Phal
Language Technologies Institute,
Carnegie Mellon University,
Pittsburgh, PA 15217
sphal@cs.cmu.edu

Abstract

This study focuses on analysing the factors that influence the performance of Parts of Speech (POS) tagging on high and low resource languages. The study begins with a preliminary analysis of the dataset and the subsequent categorization of the provided languages into high and low resource languages. Furthermore four experiments are performed using the baseline BiLSTM model which involve varying the batch size, varying the embedding dimension, varying the number of layers and changing the size of the dataset. Next a multilingual BERT model is finetuned on the provided dataset and its performance is compared against the baseline Bi-LSTM model. The findings are summarized and documented.

1 Introduction

Part of speech tagging (Voutilainen, 2012) is the process marking a word in the text with the corressponding Part of Speech tag.

2 Dataset Analysis

Upon the analysis of the dataset it was observed that four languages namely English (en), Spanish (es), Czech (cs) and Arabic (ar) had greater than 60K tokens and hence were categorized as high resource languages while Afrikaans (af), Lithuanian (lt), Armenian (hy) and Tamil (ta) had less than 60K tokens and consequently were categorized as low resource languages. All languages other than Tamil had either 18 or 19 universal POS tags.Czech had the highest number of training examples followed by English while Tamil had the least number of training examples. Furthermore Slavic had the highest number of unique tokens followed by Spanish and Arabic. In all languages NOUN was the most frequent POS tag. In addition I define a field called Inverse Token Sparsity (ITS) as #Tokens/#UniqueTokens which measures the frequency at which tokens occur in the dataset. Thus a high value of ITS would imply that the same tokens occur frequently in the dataset while a value of one would mean that each token occurs exactly once The dataset findings have been summarized in Table 2.

3 Experiments on the Baseline

The baseline here refers to a BiLSTM model with configurations, embedding dimension = 100, batch size = 128 and number of layers = 2. The baseline performance is documented in Table 1.

3.1 Varying the batch size

In the first experiment batch size is varied and its effect on the performance of high and low resource languages is compared. As can be observed from Table 3, increasing the batch size to 512 causes the test accuracy to drop for all languages. While this drop in accuracy is not prominent in high resource languages it is significant in low resource languages. Furthermore all low resource languages show between 13% - 51% drop in accuracy. On the contrary, decreasing the batch size to 32 improves the accuracy of all low resource languages while for high resource languages the accuracies either improve or reduce slightly. Interestingly a 25.3% increase in accuracy is observed in Tamil which has the least number of training examples and tokens.

3.2 Varying the embedding dimension

In this experiment we study the effect of increasing the embedding dimension. As can be observed from Table 4 increasing the embedding dimension leads to an improvement in accuracy in all low resource languages whereas no such trends are observed in high resource languages. Interestingly it turns out that this effect is visible in all languages with ITS < 15. This increase could be attributed to the fact that increasing the embedding dimensions often results in more fine grained representations leading to better performance on languages

wherein the same tokens rarely repeat. On similar lines reducing the embedding dimension causes a dip in accuracy. However this improvement in accuracy with increased dimensions comes at the expense of increased model size. On an average a 50% increase in model size is required to obtain small improvements in accuracy. This is an important consideration for applications such as on device machine learning which have limited compute and memory.

3.3 Increasing the number of layers

In this experiment I retrain the baseline with 4 layers. The observations show that in general the accuracy decreases as the depth of the network is increased. This could be attributed to overfitting The findings are summarized in Table 5.

3.4 Varying the size of the training set

In this experiment I evaluate the performance of the baseline model by using a fixed set of 200 training examples and 25%, 50% and 75% of the total number of training examples chosen uniformly at random. As can be observed from Table 6, more the data, better is the performance for both high and low resource languages. This is consistent with general machine learning theory.

4 Experiments with mBERT

In this experiment I fine tune a multilingual BERT model (Devlin et al., 2019) to measure its performance on various languages. The finetuning is performed for 5 and 10 epochs for each language with a batch size of 32 (batch size 16 was used for Arabic because of gpu memory issues) on the Nvidia RTX5000 GPU (jarvislab cloud). The BERT model had 177864975 trainable parameters. As can be observed from Table 7 mixed results are obtained on high and low resource languages. mBERT performs slightly better than the baseline on Germanic languages and shows large improvements on Dravidian languages. However on Semitic languages such as Arabic the performance is much worse than the baseline. These results are consistent with the findings of (Wu and Dredze, 2020) In general the baseline BiLSTM model has better performance on most of the languages.

5 Conclusion

In this study a preliminary analysis was performed to investigate the different factors that influence the performance of multilingual POS tagging. The results indicated that smaller batch sizes and larger embedding dimensions positively contributed to increased performance on low resource languages. Furthermore a multilingual BERT model was fine tuned and compared to the baseline BiLSTM model. Although mBERT was superior than the baseline on Germanic and Dravidian languages a hyperparameter optimized BiLSTM outperformed mBERT. Consequently the results of the fine tuned BiLSTM model were reported as summarized in Table 8.

Topic	Freq
English(en)	91.58
Spanish(es)	93.30
Czech(cs)	94.05
Arabic(ar)	94.24

Table 1: Performance of the baseline model

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

A. Voutilainen. 2012. Part-of-speech tagging. *The Oxford Handbook of Computational Linguistics*.

Shijie Wu and Mark Dredze. 2020. Are all languages created equal in multilingual BERT? In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.

Language	#Train	#Val	#Test	#Tok	#Uniq Tok	ITS	Tags	Most Freq Tag
English(en)	12543	2002	2077	204586	9863	20.74	19	NOUN (17.2%)
Spanish(es)	14187	1552	274	382436	18727	20.42	18	NOUN (18.0%)
Czech(cs)	41559	9270	10148	719317	41251	17.43	19	NOUN (24.5%)
Arabic(ar)	6174	786	704	225853	15889	14.21	18	NOUN (32.6%)
Afrikaans(af)	1315	194	425	33894	2368	14.31	18	NOUN (21.2%)
Lithuanian(cs)	2341	617	684	47605	4547	10.46	19	NOUN (30.4%)
Armenian(hy)	1975	249	278	42105	3897	10.80	19	NOUN (25.3%)
Tamil(ta)	400	80	120	6239	926	6.83	15	NOUN (28.7%)

Table 2: Analysis of the dataset

Language	Acc (32)	Acc (128)	Acc (512)
English(en)	91.58	91.58	90.81
Spanish(es)	93.76	93.30	92.91
Czech(cs)	93.93	94.05	93.77
Arabic(ar)	94.53	94.24	93.30
Afrikaans(af)	91.84	88.53	37.60
Lithuanian(cs)	77.41	75.65	48.52
Armenian(hy)	82.31	80.02	47.48
Tamil(ta)	65.14	39.84	26.46

Table 3: Effect of varying the batch sizes

Language	Params (50)	Acc (50)	Params (100)	Acc (100)	Params (200)	Acc (200)
English(en)	1077617	91.19	1621967	91.58	2710667	91.36
Spanish(es)	1520560	93.41	2508110	93.30	4483210	93.35
Czech(cs)	2647017	93.97	4760767	94.05	8988267	93.88
Arabic(ar)	1378660	94.00	2224310	94.24	3915610	94.46
Afrikaans(af)	702610	86.98	872210	88.53	1211410	91.02
Lithuanian(cs)	811817	74.51	1090367	75.65	1647467	77.17
Armenian(hy)	779317	77.53	1025367	80.02	1517467	81.38
Tamil(ta)	629739	36.02	727239	39.84	922239	48.99

Table 4: Effect of varying the embedding dimension

Language	Params (2)	Acc (2)	Params (4)	Acc (4)
English(en)	1621967	91.58	2412495	91.35
Spanish(es)	2508110	93.30	3298638	93.07
Czech(cs)	4760767	94.05	5551295	93.94
Arabic(ar)	2224310	94.24	3014838	94.34
Afrikaans(af)	872210	88.53	1662738	84.46
Lithuanian(cs)	1090367	75.65	1880895	74.27
Armenian(hy)	1025367	80.02	1815895	76.96
Tamil(ta)	727239	39.84	1517767	35.51

Table 5: Effect of increasing the number of layers

Language	Acc (200)	Acc (25%)	Acc (50%)	Acc (75%)
English(en)	23.92	87.24	89.91	90.95
Spanish(es)	35.92	89.76	91.83	92.63
Czech(cs)	33.10	89.50	92.00	93.27
Arabic(ar)	32.88	87.58	92.39	93.88
Afrikaans(af)	24.85	39.41	73.99	83.73
Lithuanian(cs)	30.73	48.25	62.28	73.58
Armenian(hy)	37.03	47.85	67.07	76.35
Tamil(ta)	34.61	32.04	24.61	37.78

Table 6: Effect of changing the size of the dataset

Language	Train (5)	Val (5)	Test (5)	Train (10)	Val (10)	Test (10)
English(en)	97.32	92.48	92.02	98.78	92.69	91.98
Spanish(es)	97.23	94.85	92.51	98.89	94.50	92.74
Czech(cs)	85.42	81.91	81.61	90.50	80.81	81.58
Arabic(ar)	47.15	45.61	47.38	47.70	45.43	47.35
Afrikaans(af)	91.93	87.42	88.62	96.46	87.95	87.98
Lithuanian(cs)	75.36	69.08	65.90	81.92	69.09	66.16
Armenian(hy)	74.08	68.31	69.10	82.21	69.42	69.02
Tamil(ta)	57.45	55.12	56.13	64.58	54.79	56.49

Table 7: Performance of multilingual BERT with 5 and 10 epochs

Language	Train	Val	Test
English(en)	98.52	91.41	91.72
Spanish(es)	98.56	93.84	93.39
Czech(cs)	99.32	94.29	94.13
Arabic(ar)	98.90	93.78	94.72
Afrikaans(af)	97.40	91.69	92.22
Lithuanian(cs)	94.68	81.32	77.65
Armenian(hy)	94.03	85.18	82.44
Tamil(ta)	81.25	70.70	67.40

Table 8: Performance of fine tuned baseline model embedding dimension = 200 and batch size = 32